Industrial Visual Anomaly Detection in Robotics: Methods, Datasets, and Deployable System Architectures

Thanh-Hai Tran[1,2] and Xuan-Bach Le[1,2⋆]

[1] Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology (HCMUT), Ho Chi Minh City, Vietnam
[2] Vietnam National University Ho Chi Minh City, Ho Chi Minh City, Vietnam
{tthai.sdh241,lexuanbach}@hcmut.edu.vn

Abstract. As collaborative robots (cobots) are increasingly integrated into unstructured environments, intelligent safety systems capable of detecting hazards without relying on extensive labeled data become necessary. This survey reviews deep learning methods for Industrial Visual Anomaly Detection (IVAD), emphasizing their role in robotic safety monitoring. Unlike static quality inspection, robotic IVAD must function in real time, handle dynamic scenes, and detect open-set anomalies. Existing approaches are categorized into reconstruction-based, embedding-based, and vision–language model (VLM) paradigms. Relevant benchmarks and robotics-specific datasets are examined, highlighting challenges such as motion, viewpoint shifts, and multimodal sensing. Strong benchmark scores do not carry over to robotic deployments, where dynamic scenes, latency limits, and semantic risks apply. This survey treats IVAD as a safety-critical system component—one that requires edge efficiency, low latency, and integration with middleware such as ROS2. The survey closes by mapping open problems and practical design directions for practitioners.

Keywords: Industrial Visual Anomaly Detection · Robotics · Unsupervised Learning · Vision–Language Models · Edge AI · ROS2

1 Introduction

High-Mix Low-Volume (HMLV) manufacturing has redefined industrial robotics, shifting robots from isolated cells to shared workspaces with humans [1]. Operating in shared spaces requires safety systems that can reason about context, not just detect motion: capabilities that traditional binary sensors like light curtains and mats lack [2]. These systems cannot distinguish between true hazards and routine actions, leading to false alarms or missed threats.

One practical barrier is that failure data is scarce [3,4]: in optimized factories, events like collisions or unexpected object appearances are rare. Collecting

⋆ Corresponding author

labeled anomalies for supervised models is impractical and raises ethical concerns. Worse, many hazards are open-set, appearing for the first time during deployment.

Unsupervised Anomaly Detection (UAD) sidesteps this [5,6] by learning what "normal" looks like and flagging deviations. Two main approaches have emerged: fast, lightweight models for edge deployment [7,8], and Vision–Language Models (VLMs) for detecting semantic risks like unsafe tool use [9,10].

Current surveys focus largely on static inspection tasks under controlled conditions, often overlooking challenges critical to robotic safety: dynamic scenes, real-time constraints, multimodal inputs, and semantic anomalies. As a result, the gap between academic benchmarks and practical deployment remains poorly addressed. This survey addresses this gap through three key contributions:

- The paper introduces a robotics-specific taxonomy covering structural, logical, semantic, and temporal deviations (Section 2).
- Methods are evaluated not just on accuracy but on latency, memory usage, and edge compatibility (Section 3).
- Section 4 and Section 5 discuss how to wire anomaly detection into a ROS2 stack and identify open engineering challenges.

Throughout, the paper treats IVAD as an engineering problem: not just a matter of detection accuracy, but of building systems that work safely under hardware and time constraints. Section 2 presents the taxonomy and datasets; Section 3 reviews methods with emphasis on deployment trade-offs; Section 4 covers system design and integration; Section 5 maps research gaps and future directions; Section 6 concludes.

2 Anomaly Taxonomy and Datasets

Without labeled data, detection depends on knowing what normal looks like, which requires a clear account of what kinds of anomalies can occur. This section proposes a four-part taxonomy reflecting the diverse safety and performance deviations seen in real-world systems.

2.1 Taxonomy of Robotic Anomalies

Figure 1 presents four key types of robotic anomalies, grouped by how and where deviations occur. The four categories expose why no single detection method can cover all failure modes.

Structural anomalies. These involve visible deviations on object surfaces or the robot itself [11,12], and are the most studied in industrial visual inspection. Common cases include scratches, cracks, oil spills, or debris in the workspace. Since they affect local appearance and are frame-independent, they suit methods based on reconstruction or embeddings. Missing them typically degrades product quality or accelerates equipment wear, but does not put people at immediate risk.

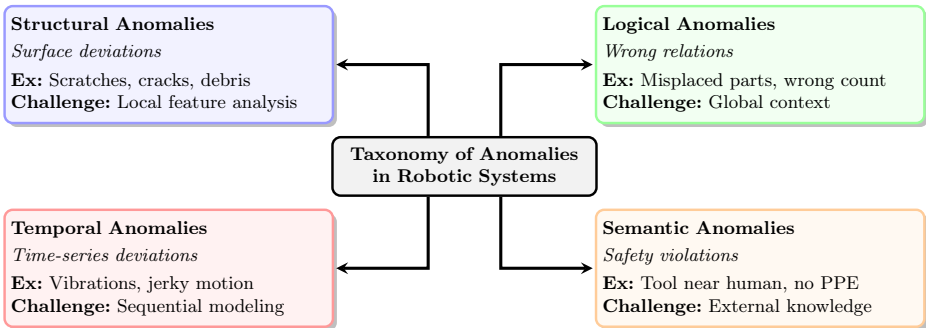

Fig. 1: Taxonomy of anomalies in robotic systems.

Logical anomalies. Logical anomalies occur when individual elements appear visually correct but collectively violate spatial arrangements, quantities, or task sequences [13]. Examples include a bolt placed in the wrong bin or a robot fastening five screws when only four are required. Because these violations often involve correct-looking parts in incorrect configurations, they typically evade patch-based detection methods that focus on local features. Catching them requires reasoning over global context, not just local texture, and a learned sense of what a correct assembly looks like [14]. A misplaced bolt or extra screw can pass through assembly unnoticed and only surface as a field failure.

Semantic anomalies. These are situations that look visually normal but break a safety rule or common-sense norm, detectable only with contextual knowledge [15]. Examples include a robot moving a tool toward a person or a worker without protective gear in a hazardous area [2]. Detecting them requires an understanding of human–robot interaction, tool usage, and safety policies, not just visual patterns. Unlike the other types, failing to detect a semantic risk can cause direct physical harm, even when the scene looks completely normal.

Temporal and kinematic anomalies. These occur when a robot's motion or state changes abnormally over time [3], such as through vibrations, jerky movements, sudden stops, or trajectory drift. Unlike visual anomalies, they require sequence-level analysis using recurrent or time-series models. Because they often signal emerging mechanical faults, early detection is critical for predictive maintenance and system reliability.

These four types (structural, logical, semantic, and temporal) cover the main ways a robotic operation can fail visually without triggering standard alerts. Each demands a different detection strategy; no single method handles all four well.

2.2 Anomaly Detection Datasets

Dataset choice shapes what a model can learn and what gaps in performance go undetected. In practice, existing IVAD resources can be broadly categorized by domain (industrial inspection versus robot-specific settings) and by modality, ranging from single-image inputs to multi-view and multimodal sensor data.

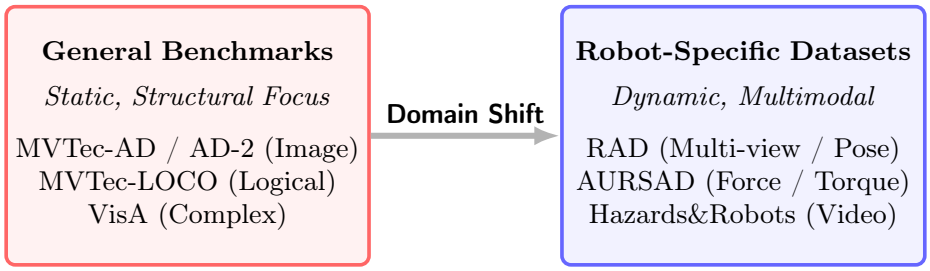

Fig. 2: Comparison of general industrial benchmarks and robot-specific anomaly datasets.

Figure 2 contrasts two families of datasets that have shaped anomaly detection research. General industrial benchmarks such as MVTec-AD and VisA provide precise pixel-level annotations under controlled lab conditions, well suited to structural anomaly evaluation. Robot-specific datasets, by contrast, reflect the dynamic and multimodal nature of real robotic deployments, capturing motion-induced variation, viewpoint changes, and sensor fusion. The following subsections examine representative datasets from each family, highlighting their strengths and the gaps that remain open for real-world robotic systems.

General industrial benchmarks. MVTec-AD [11] and VisA [16] are the standard benchmarks for visual anomaly detection. Both provide high-quality images of surface defects under controlled lab conditions, but scores are now saturated and newer methods are hard to distinguish.

To address this limitation, MVTec-LOCO-AD [13] introduces logical anomalies, in which scenes remain visually plausible but violate compositional or structural constraints. Nevertheless, MVTec-AD, LOCO, and VisA all assume static scenes with fixed viewpoints and uniform lighting. MVTec-AD-2 [17] partially relaxes these assumptions by introducing transparent materials and illumination variation, exposing significant performance drops and revealing the fragility of many methods under real-world distribution shifts.

Robot-specific datasets. In contrast to static benchmarks, robot-specific datasets are designed to capture the dynamic conditions encountered during real robotic operation, including motion, viewpoint changes, and multimodal sensing. The RAD dataset provides multi-view visual observations from a Franka Emika robotic arm and incorporates realistic pose noise, making it well suited for evaluating view-invariant anomaly detection under deployment-like conditions [18]. However, RAD primarily emphasizes structural and logical anomalies, offering limited coverage of higher-level semantic safety violations.

Other datasets focus on task-specific or non-visual modalities. AURSAD captures force, torque, current signals during screw-driving tasks with a UR3e robot, effectively highlighting procedural errors such as missing screws or collisions [19]. Voraus-AD similarly provides joint and motor time-series data for pick-and-place tasks [20], supporting temporal anomaly detection in the absence of visual input.

Hazards&Robots focuses on mobile robots, providing large-scale video data of navigation in dynamic, human-involved environments [21]. It captures safety-critical interactions well, but precise localization and fine-grained structural anomalies remain difficult due to motion noise and shifting context.

Existing datasets do not capture the full complexity of robotic anomaly detection: dynamic viewpoints, semantic safety violations, multimodal inputs, and real-time constraints. General benchmarks cover structural flaws but miss temporal and semantic risks; robot-specific datasets handle dynamics but often lack precise safety context and localization. Until both concerns (annotation quality and deployment realism) are addressed in the same benchmark, evaluation results will remain optimistic.

2.3 Empirical Distribution Analysis

To assess the practical prevalence of different anomaly types, three key robotic datasets are analyzed: RAD [18] (4,287 images), AURSAD [19] (approximately 2,000 samples), and Hazards&Robots [21] (324,408 video frames).

Across these datasets, structural anomalies like defects and workspace disturbances account for about 67% of labeled cases in the original dataset annotations. While state-of-the-art methods now exceed 95% accuracy on such tasks [7,22], logical anomalies remain challenging with typical AUROC scores ranging from 75–86% on benchmarks like MVTec-LOCO [22,7].

Recent semantic-aware methods like SALAD [23] lead in logical anomaly detection by modeling object composition and spatial relations. While they outperform appearance-based approaches, they remain computationally heavy and untested in real-time or highly dynamic robotic settings.

Anomaly patterns vary by application. In manipulation tasks (RAD, AURSAD), most anomalies (78%) stem from product defects and process errors. In mobile settings like Hazards&Robots, 62% involve workspace hazards and human intrusions, often tied to semantic safety. Manufacturing settings call for logical reasoning over task structure; mobile settings demand semantic awareness of the workspace.

3 Deep Learning Approaches

Three generations of IVAD methods have emerged: reconstruction, embedding, and semantic, each resting on different assumptions about what anomalies look like and what can go wrong in practice.

3.1 Reconstruction-based Methods

Reconstruction-based methods assume that models trained only on normal data will poorly reconstruct anomalies, using reconstruction error as the anomaly signal. This relies on the idea that anomalies are hard to reconstruct, an assumption that often fails in structured or repetitive scenes.

Early methods use convolutional autoencoders to compress and reconstruct images, but these models often generalize too well, reproducing anomalies and missing them. GAN-based methods such as AnoGAN [24] and its faster variant f-AnoGAN [25] attempt to reconstruct images through a learned latent space, using residual error for detection. Memory-Augmented Autoencoders (MemAE) address generalization by forcing reconstruction through a fixed set of memory slots, limiting generalization to known patterns [26].

For sequential robot data, LSTM-VAE models extend reconstruction to time-series by modeling distributions over joint trajectories, forces, or velocities. Deviations flag potential issues like collisions or wear [20,27]. While interpretable, these methods often struggle in cluttered scenes and are ill-suited for logical or semantic anomalies.

3.2 Embedding-based Methods

Embedding-based methods skip pixel reconstruction entirely, working directly in feature space with pre-trained network embeddings. The result is better robustness to lighting variation and texture shift, at the cost of interpretability and any semantic understanding.

Knowledge distillation. Teacher–student frameworks train a lightweight student to mimic a frozen teacher's features on normal data, with anomalies flagged by feature mismatch at inference. EfficientAD follows this approach, enabling real-time deployment with sub-3ms latency [7]. Student-Teacher Feature Pyramid Matching (STFPM) [28] extends this by matching multi-scale pyramid features, improving localization quality. Reverse Distillation (RD4AD) [29] takes the opposite direction, feeding the teacher's one-class embedding into a student decoder, which improves compactness and anomaly sensitivity. However, patch-based distillation designs limit global context, reducing effectiveness on logical anomalies.

Normalizing flows. Flow-based methods like FastFlow [30] and CFLOW-AD [31] transform complex features into Gaussian space using invertible mappings, detecting anomalies via low likelihood. CFLOW-AD adds conditional vector quantization for improved localization. The trade-off is higher latency compared to distillation-based models, though localization quality tends to be better.

Memory bank methods. PatchCore detects anomalies by nearest-neighbor comparison of patch-level features against a memory bank of normal samples [22]. PaDiM [32] takes a complementary approach by fitting multivariate Gaussian distributions per patch position, enabling localization with fewer parameters. SimpleNet [33] introduces a lightweight anomaly feature generator (adding noise to normal features to synthesize hard negatives), achieving 99.6% AUROC on MVTec-AD at 77 FPS. In static settings they work well, but the memory bank grows with scene complexity, posing a practical barrier for real-time robotic deployment.

Semantic-aware anomaly detection. Recent methods focus on anomalies that look visually normal but break object-level structure. SALAD addresses

this through a three-branch architecture (local appearance, object composition maps, and global context), rather than relying on raw patch features alone [23]. This design captures spatial and semantic part relationships that single-branch distillation models miss, which explains both its stronger logical anomaly performance and its higher inference cost of ∼65 ms. While this improves detection of logical anomalies, the latency precludes real-time robotic deployment.

Embedding-based methods are fast and accurate on appearance-based tasks, but they cannot reason about intent or context, a gap that becomes critical when the anomaly is semantic rather than structural.

3.3 Vision–Language Models (VLMs)

VLMs work differently from prior methods: rather than flagging pixels that look wrong, they assess whether a scene makes sense given a textual description, enabling zero-shot detection but at significant computational cost.

Models like CLIP [34] enable prompt-based anomaly detection by aligning images with textual descriptions of normal and unsafe conditions. WinCLIP [35] extends this for zero-shot industrial inspection by using window-level compositional prompts, achieving 91.8% AUROC on MVTec-AD without any task-specific training. CLIP-ADA [36] adapts CLIP for industrial inspection using anomaly-aware prompts, while VadCLIP adds temporal context for video [10]. AnomalyGPT brings this approach to industrial use with explanatory outputs [9], and AnomalyRuler integrates rule-based supervision for more controllable detection [37]. Recent work on VLMs for laboratory robots [38] has shown early results on semantic safety monitoring, but inference times are nowhere near the sub-50ms needed for real-time response [9,10].

3.4 Performance Comparison and Design Implications

Table 1 lays out representative IVAD methods by paradigm, logical anomaly handling, latency, and deployment feasibility. The pattern is clear: models that handle logical anomalies well pay for it in inference cost and deployment constraints. Note that latency values are as reported in the respective source papers on server-grade GPUs (GPU model varies per paper); edge deployment figures may differ.

EfficientAD-S reports 85.8% image-level AUROC on MVTec-LOCO, compared with PatchCore (75.8%) in the same summary table [7]; the medium variant (EfficientAD-M) reaches 90.7%, narrowing the gap with semantic models like SALAD. At the same time, reported performance on structural benchmarks is typically higher (>94% AUROC) [7], consistent with these models relying primarily on local appearance cues with limited global reasoning. PatchCore remains strong on static benchmarks but can be brittle when the viewpoint or background changes substantially, as is common in mobile or eye-in-hand settings [21].

Table 1: Comparison of representative IVAD approaches.

Method	Paradigm	Logical AD	Latency	Deployment
EfficientAD [7]	Distillation	Moderate	2.2 ms[b]	Real-time
FastFlow [30]	Flow-based	Limited	∼17 ms	Near real-time
PatchCore [22]	Memory Bank	Limited	∼32 ms[a]	Offline / Edge
SALAD [23]	Semantic Embedding	Strong	∼65 ms[c]	Offline / Near RT
AnomalyGPT [9]	VLM	Strong	>500 ms	Cloud-only

[a] PatchCore speed varies with coreset sampling ratio; 224.4 ms reported at full scale [23].

[b] EfficientAD measured on RTX A6000 [7]; on A100, same model runs at 6.2 ms [23].

[c] SALAD measured on NVIDIA A100 [23].

No single method meets all three requirements at once. In practice, this points to hybrid designs: a fast local detector handles the real-time loop, while a heavier reasoning module runs asynchronously outside the control path.

4 System Design and Deployment Considerations

A detector that scores well on benchmarks is only part of the story; it must also run within compute budgets, fit into a middleware stack, and respond fast enough to be safety-relevant. In practice, a slightly less accurate model that runs at 30 FPS on a Jetson device is more useful than a state-of-the-art model that requires a cloud GPU.

4.1 Edge Computing Platforms

Cloud-based processing adds latency and network risk, both unacceptable for a safety-critical response path. Anomaly detection must run at the edge, i.e., onboard or nearby. Embedded GPU platforms such as NVIDIA Jetson devices are widely used for this purpose: entry-level platforms like Jetson Nano handle lightweight models in real time, while Jetson Orin series supports higher-throughput inference with optimized models such as EfficientAD [7,39].

Model optimization is not optional here. TensorRT quantization from FP32 to FP16 or INT8 typically delivers 2–5× speedup with minimal accuracy loss [39], often enough to bring a complex model within budget on embedded hardware. The detector must finish quickly enough to leave room for the planner and actuator within the safety window.

4.2 ROS2 Integration Architecture

The Robot Operating System 2 (ROS2) offers a standardized, modular middleware for real-time robotic systems [40]. In practice, anomaly detection runs as distributed ROS2 nodes using a publish–subscribe model. Unlike idealized

benchmarks, real deployments must factor in message passing, sensor sync, and processing overhead, all of which affect overall system latency.

Figure 3 illustrates a reference deployment architecture, adapted from collaborative human-robot safety frameworks [2,40]. A typical deployment includes a perception node that captures visual data, and an anomaly detection node that performs lightweight inference (e.g., EfficientAD), optionally accelerated by engines like OpenVINO [41]. Detected anomalies are published for evaluation by a safety supervision node, which coordinates mitigation actions via the robot control node [2]. The full pipeline must stay within the safety response window; any node that stalls breaks the chain.

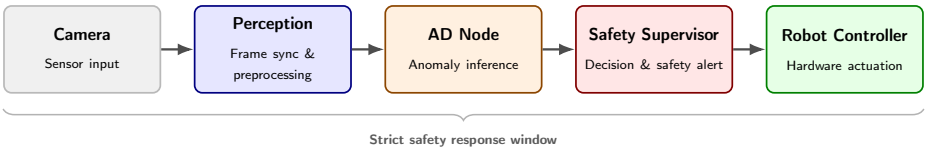

Fig. 3: Reference ROS2-based IVAD deployment architecture.

In parallel, a logging and monitoring node records anomaly events with synchronized sensor and system data for traceability, debugging, and post-incident analysis. Keeping the nodes separate means the detector, supervisor, and controller can be updated or replaced independently, which is useful as requirements change across deployment contexts.

5 Research Gaps and Future Directions

The gaps fall into three areas: limitations in semantic and logical reasoning, constraints from real-time and edge deployment, and weaknesses in current evaluation methodology.

5.1 Identified Research Gaps

Logical anomaly detection. State-of-the-art IVAD methods perform well on structural defects using local feature comparisons [7,22], but fall short on logical anomalies where components appear normal yet violate spatial or assembly rules [13]. Patch-based models lack global context, and EfficientAD provides only limited reasoning through an auxiliary autoencoder branch [7]. Semantic-aware approaches like SALAD improve logical anomaly detection by modeling object composition [23], but none yet run in real-time on deployed robotic hardware.

Sim-to-real transfer. Due to safety constraints, generating real anomalies at scale is impractical, making simulation an appealing substitute [42,43]. Yet, models trained in simulated environments often fail in deployment because of

domain gaps in lighting, textures, and physics. Unlike supervised tasks, sim-to-real transfer for unsupervised anomaly detection is less explored, hindered by the absence of labeled target data.

Explainability. ISO/TS 15066 demands transparent, auditable decisions. Most IVAD methods output heatmaps but lack clarity on *why* an anomaly is detected [7,22]. Vision–Language Models offer explanations but are too slow for real-time use [9,10]. Getting both speed and interpretability in a single deployable model is the core unsolved problem here.

Dynamic background handling. Most IVAD methods are designed for static backgrounds and fixed cameras, which works well for conveyor belts but not for mobile or eye-in-hand robots [21]. In practice, any background motion triggers false positives, making these methods unreliable outside the lab.

Continual learning at the edge. Industrial settings evolve as products and processes change, requiring robots to adapt without catastrophic forgetting [44] or centralized retraining. No stable, edge-compatible approach to continual learning currently exists for this setting.

Multi-robot coordination anomalies. In collaborative settings, anomalies can stem from coordination issues like deadlocks or resource conflicts rather than individual faults. Detecting these requires joint reasoning across agents, something current IVAD methods, built for single robots, typically overlook.

Real-time VLM inference. VLMs provide rich semantic reasoning but are too slow for safety-critical tasks requiring sub-50ms responses [9,10]. Distilling their knowledge into fast, edge-ready models without losing reasoning ability is a key research priority [8].

Uncertainty quantification. Anomaly detection systems should offer calibrated confidence to enable risk-aware decisions. Bayesian approximation methods [45] and evidential deep learning show promise, but are rarely applied in IVAD, especially in multimodal settings where sensor reliability varies [5].

Closing these gaps is what separates benchmark progress from systems that work in the field. Accuracy on clean benchmarks is no longer the bottleneck; the open problems lie in reasoning, efficiency, and robustness under deployment conditions.

5.2 Proposed Future Directions

Hybrid VLM–lightweight architectures. A hybrid design is proposed where a VLM runs offline or intermittently to generate pseudo-labels, safety rules, or synthetic data for training lightweight models deployable at the edge [8]. This distillation-based approach offers a practical path to combining rich semantic reasoning with real-time performance.

Multimodal fusion with uncertainty quantification. Integrating vision with audio, force/torque, and proprioceptive sensing through evidential or probabilistic frameworks can reduce false positives and yield interpretable confidence estimates for each modality's contribution [5].

Few-shot logical anomaly detection. Given the scarcity of logical anomalies, meta-learning methods like Model-Agnostic Meta-Learning, paired with syn-

thetic data generation (e.g., ComGEN [46]), can offer a path to fast adaptation with minimal supervision. Registration-based approaches such as RegAD [47] and unified frameworks like UniFormaly [48] further address this challenge by enabling category-agnostic detection across diverse anomaly types with few reference samples.

Active learning with human-in-the-Loop. By engaging human operators only for uncertain cases, active learning can improve models efficiently with minimal disruption. Key challenges lie in when to query, who to ask, and how to integrate feedback effectively.

Cross-domain generalization. Learning domain-invariant features can enable anomaly detectors to transfer across varying production settings, reducing the need for frequent retraining [42,43].

Complete hybrid system implementation. The field needs full-stack implementations that combine fast edge detection, accurate localization, and cloud-based semantic reasoning. A ROS2-compatible reference system would give practitioners a concrete starting point.

Together, these directions address the practical barriers that keep IVAD from leaving the lab. Safe human–robot collaboration requires anomaly detection that is integrated, not standalone: a detector that fits into a control loop, explains its outputs, and degrades gracefully under uncertainty.

6 Conclusion

This survey reviewed deep learning methods for industrial visual anomaly detection in robotics, emphasizing approaches that work without large labeled anomaly datasets. A robotics-focused taxonomy is proposed (structural, logical, semantic, and temporal) that highlights how each anomaly type demands distinct perceptual and reasoning capabilities.

Three patterns stand out from the review. Strong benchmark results often fail to translate into real-world performance, especially in dynamic robotic settings. Detecting safety-critical anomalies depends more on semantic understanding than on local feature quality. Real-time and resource-efficient design cannot be left as an afterthought given the strict demands of safety-critical deployment.

These challenges (unsupervised logical anomaly detection, robust sim-to-real transfer, evaluation under dynamic, interactive conditions) highlight the disparity between lab benchmarks and real-world deployment. Until these are resolved, IVAD will keep working in the lab but not in the field.

Hybrid systems look like the most viable path forward: a fast local detector handles the real-time loop, while a heavier reasoning module operates asynchronously. Multimodal sensing (combining vision, proprioception, and environmental inputs) will further improve robustness and uncertainty awareness. The harder challenge ahead is not improving AUROC scores but building systems that run reliably within latency and compute budgets.

Disclaimer. The authors used AI-based tools as an assisted editing tool, to improve spelling, grammar, clarity, and readability during manuscript preparation.

After using this tool, the authors carefully reviewed, edited the text and take full responsibility for the final content.

Acknowledgments

We acknowledge Ho Chi Minh City University of Technology (HCMUT), VNUHCM for supporting this study.

References

1. Shukla, V., Shukla, A., S K, S.P., Shukla, S.: A systematic survey: role of deep learning-based image anomaly detection in industrial inspection contexts. Front. Robot. AI **12**, 1554196 (2025)
2. Amaya-Mejía, L.M., Duque-Suárez, N., Jaramillo-Ramírez, D., Martinez, C.: Vision-based safety system for barrierless human-robot collaboration. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7331–7336. IEEE (2022)
3. Yahya, M.A., Moya, A.R., Ventura, S.: Deep learning for multivariate time series anomaly detection: an evaluation of reconstruction-based methods. Artif. Intell. Rev. **58**, 400 (2025)
4. Chirayil Nandakumar, S., Mitchell, D., Erden, M.S., Flynn, D., Lim, T.: Anomaly detection methods in autonomous robotic missions. Sensors **24**(4), 1330 (2024)
5. Yang, Y., Zhao, J., Xu, X., Cao, K., Yuan, S., Xie, L.: Unsupervised anomaly detection for autonomous robots via Mahalanobis SVDD with audio-IMU fusion. arXiv preprint arXiv:2505.05811 (2025)
6. Kang, T., You, B.J., Park, J., Lee, Y.: A real-time anomaly detection method for robots based on a flexible and sparse latent space. Eng. Appl. Artif. Intell. **158**, 111310 (2025)
7. Batzner, K., Heckler, L., König, R.: EfficientAD: Accurate visual anomaly detection at millisecond-level latencies. In: 2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 127–137. IEEE (2024)
8. Barusco, M., Borsatti, F., Dalle Pezze, D., Paissan, F., Farella, E., Susto, G.A.: PaSTe: Improving the efficiency of visual anomaly detection at the edge. In: Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 4026–4035 (2025)
9. Gu, Z., Zhu, B., Zhu, G., Chen, Y., Tang, M., Wang, J.: AnomalyGPT: Detecting industrial anomalies using large vision-language models. In: Proc. AAAI Conf. on Artificial Intelligence **38**(3), pp. 1932–1940 (2024)
10. Wu, P., Zhou, X., Pang, G., et al.: VadCLIP: Adapting vision-language models for weakly supervised video anomaly detection. In: Proc. AAAI Conf. on Artificial Intelligence **37**(2), pp. 2726–2734 (2023)
11. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: MVTec AD – A comprehensive real-world dataset for unsupervised anomaly detection. In: Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 9592–9600 (2019)
12. Bergmann, P., Batzner, K., Fauser, M., Sattlegger, D., Steger, C.: The MVTec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection. Int. J. Comput. Vis. **129**(4), 1038–1059 (2021)
13. Bergmann, P., Batzner, K., Fauser, M., Sattlegger, D., Steger, C.: Beyond dents and scratches: logical constraints in unsupervised anomaly detection and localization. Int. J. Comput. Vis. **130**(4), 947–969 (2022)

14. Patra, S., Ben Taieb, S.: Revisiting deep feature reconstruction for logical and structural industrial anomaly detection. arXiv preprint arXiv:2410.16255 (2024)
15. Brunke, L., Zhang, Y., Römer, R., et al.: Semantically safe robot manipulation: from semantic scene understanding to motion safeguards. IEEE Robot. Autom. Lett. **PP**, 1–8 (2025)
16. Zou, Y., Jeong, J., Pemula, L., Zhang, D., Dabeer, O.: Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In: Proc. European Conf. on Computer Vision (ECCV), pp. 392–408 (2022)
17. Heckler-Kram, L., Neudeck, J.H., Scheler, U., König, R., Steger, C.: The MVTec AD 2 dataset: advanced scenarios for unsupervised anomaly detection. Int. J. Comput. Vis. **134**(4), 2743–2760 (2026)
18. Zhou, K., Chang, X., Kim, T., et al.: RAD: A dataset and benchmark for real-life anomaly detection with robotic observations. arXiv preprint arXiv:2410.00713 (2026)
19. Leporowski, B., Tola, D., Hansen, C., Iosifidis, A.: AURSAD: Universal robot screwdriving anomaly detection dataset. arXiv preprint arXiv:2102.01409 (2021)
20. Brockmann, J.T., Rudolph, M., Rosenhahn, B., Wandt, B.: The voraus-AD dataset for anomaly detection in robot applications. IEEE Trans. Robot. **40**, 438–451 (2024)
21. Mantegazza, D., Xhyra, A., Gambardella, L.M., Giusti, A., Guzzi, J.: Hazards&Robots: a dataset for visual anomaly detection in robotics. Data Brief **48**, 109264 (2023)
22. Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., Gehler, P.: Towards total recall in industrial anomaly detection. In: Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 14312–14322 (2022)
23. Fučka, M., Zavrtanik, V., Skočaj, D.: SALAD – Semantics-aware logical anomaly detection. In: Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV), pp. 1–11 (2025)
24. Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: Proc. Int. Conf. on Information Processing in Medical Imaging (IPMI), pp. 146–157 (2017)
25. Schlegl, T., Seeböck, P., Waldstein, S.M., Langs, G., Schmidt-Erfurth, U.: f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks. Med. Image Anal. **54**, 30–44 (2019)
26. Gong, D., Liu, L., Vuong, L., et al.: Memorizing normality to detect anomaly: memory-augmented deep autoencoder for unsupervised anomaly detection. In: Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV), pp. 1705–1714 (2019)
27. Park, D., Hoshi, Y., Kemp, C.C.: A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder. IEEE Robot. Autom. Lett. **3**(3), 1544–1551 (2018)
28. Wang, G., Han, S., Ding, E., Huang, D.: Student-teacher feature pyramid matching for anomaly detection. In: Proc. British Machine Vision Conf. (BMVC), Article 349, pp. 1–11 (2021)
29. Deng, H., Li, X.: Anomaly detection via reverse distillation from one-class embedding. In: Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 9737–9746 (2022)
30. Yu, J., Zheng, Y., Wang, X., et al.: FastFlow: unsupervised anomaly detection and localization via 2D normalizing flows. arXiv preprint arXiv:2111.07677 (2021)

31. Gudovskiy, D., Ishizaka, S., Kozuka, K.: CFLOW-AD: real-time unsupervised anomaly detection with localization via conditional normalizing flows. In: Proc. IEEE/CVF Winter Conf. on Applications of Computer Vision (WACV), pp. 98–107 (2022)
32. Defard, T., Setkov, A., Loesch, A., Audigier, R.: PaDiM: a patch distribution modeling framework for anomaly detection and localization. In: Proc. Int. Conf. on Pattern Recognition (ICPR) Workshops, pp. 475–489 (2021)
33. Liu, Z., Zhou, Y., Xu, Y., Wang, Z.: SimpleNet: a simple network for image anomaly detection and localization. In: Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 20402–20411 (2023)
34. Radford, A., Kim, J.W., Hallacy, C., et al.: Learning transferable visual models from natural language supervision. In: Proc. Int. Conf. on Machine Learning (ICML), pp. 8748–8763 (2021)
35. Jeong, J., Zou, Y., Kim, T., Zhang, D., Ravichandran, A., Dabeer, O.: WinCLIP: zero-/few-shot anomaly classification and segmentation. In: Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 19606–19616 (2023)
36. Cai, Y., He, X., Liang, D., Tong, A., Bai, X.: Anomaly detection by adapting a pre-trained vision language model. arXiv preprint arXiv:2403.09493 (2024)
37. Yang, Y., Lee, K., Dariush, B., Cao, Y., Lo, S.Y.: Follow the rules: reasoning for video anomaly detection with large language models. In: Proc. European Conf. on Computer Vision (ECCV), LNCS 15139, pp. 312–329 (2025)
38. Lin, S., Wang, C., Ding, X., et al.: A VLM-based method for visual anomaly detection in robotic scientific laboratories. arXiv preprint arXiv:2506.05405 (2025)
39. NVIDIA: TensorRT: High performance deep learning inference. Technical Documentation (2023). https://developer.nvidia.com/tensorrt
40. Antmicro: Dedicated ROS 2 nodes for AI-enabled computer vision tasks (2023). https://antmicro.com/blog/2023/08/ros2-nodes-for-computer-vision
41. Akcay, S., Ameln, D., Vaidya, A., Lakshmanan, B., Ahuja, N., Genc, U.: Anomalib: a deep learning library for anomaly detection. In: Proc. IEEE Int. Conf. on Image Processing (ICIP), pp. 1706–1710 (2022)
42. James, S., Wohlhart, P., Kalakrishnan, M., et al.: Sim-to-real via sim-to-sim: data-efficient robotic grasping via randomized-to-canonical adaptation networks. arXiv preprint arXiv:1812.07252 (2019)
43. Zhang, Z., Zhao, Z., Zhang, X., Sun, C., Chen, X.: Industrial anomaly detection with domain shift: a real-world dataset and masked multi-scale reconstruction. arXiv preprint arXiv:2304.02216 (2023)
44. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., et al.: Overcoming catastrophic forgetting in neural networks. Proc. Natl. Acad. Sci. U.S.A. **114**(13), 3521–3526 (2017)
45. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: representing model uncertainty in deep learning. In: Proc. Int. Conf. on Machine Learning (ICML), pp. 1050–1059 (2016)
46. Tong, X., Chang, Y., Zhao, Q., et al.: Component-aware unsupervised logical anomaly generation for industrial anomaly detection. In: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 16722–16729 (2025)
47. Huang, C., Guan, H., Jiang, A., Zhang, Y., Spratling, M., Wang, Y.F.: Registration based few-shot anomaly detection. arXiv preprint arXiv:2207.07361 (2022)
48. Lee, Y., Lim, H., Jang, S., Yoon, H.: UniFormaly: towards task-agnostic unified framework for visual anomaly detection. arXiv preprint arXiv:2307.12540 (2023)